

<https://periodicos.utfpr.edu.br/recit>

Avaliando a ferramenta jgoose utilizando princípios da engenharia de software experimental

RESUMO

Avaliar ferramentas computacionais bem como novas técnicas ou processos exige cuidados especiais tanto no planejamento quanto na execução e análise no experimento realizado. Neste contexto, a engenharia de software experimental apresenta algumas metodologias que podem ser úteis na condução de experimentos. Mais especificamente, neste artigo relata-se os resultados obtidos como o uso das Metodologias Survey e Experimento Controlado na experimentação da ferramenta JGOOSE, a qual foi desenvolvida pelo nosso grupo de pesquisa e permite derivar automaticamente casos de uso UML a partir de modelos organizacionais i*. A metodologia Survey, a qual permite obter tanto uma avaliação qualitativa quanto quantitativa da referida ferramenta, foi aplicada com acadêmicos de uma disciplina de Engenharia de Software como também com alguns especialistas na área. Já o experimento controlado, o qual permite um maior controle sobre as variáveis envolvidas, foi aplicado em um grupo de participantes envolvendo acadêmicos e profissionais da área.

PALAVRAS-CHAVE: Ferramenta Computacional, i*(istar), Survey, Experimento Controlado

INTRODUÇÃO

Ferramentas que permitem a modelagem de ambientes organizacionais ajudam os engenheiros de requisitos tanto no entendimento desse ambiente quanto dos requisitos de sistemas computacionais inseridos nesse ambiente. Nesse contexto, uma das ferramentas que tem recebido destaque é a JGOOSE (*Java Goal into Object Oriented Standard Extension*) (Vicente, 2009) (Les, 2016). Nesta ferramenta, os modelos organizacionais são construídos via framework i* (Yy, 1995) e a partir destes, permite-se gerar casos de uso UML (Booch et al., 2005). O framework i* tem como foco a modelagem das intencionalidades dos atores na organização. Para esse fim, dois modelos são propostos: Modelo de Dependências Estratégicas (SD) e Modelo de Razões Estratégicas (SR). Já casos de uso UML são amplamente conhecidos e utilizados para modelar requisitos funcionais. Contudo, a JGOOSE precisa ser adequadamente avaliada. Para esse fim, é importante utilizar métodos propostos pela Engenharia de Software Experimental. Entre as principais metodologias advindas da Engenharia de Software Experimental (ESE) pode-se destacar *Survey*, Estudo de Caso e Experimento (Wholin et al., 2000). Cabe ressaltar que este artigo visa relatar os estudos realizados para avaliar a JGOOSE utilizando as metodologias *Survey* e Experimento Controlado.

A seguir apresenta-se na seção 2 uma breve descrição da ferramenta JGOOSE bem como das principais características das metodologias propostas pela ESE além da justificativa do uso das metodologias escolhidas para avaliar a ferramenta JGOOSE. Na seção 3 são apresentados os resultados advindos do uso da metodologia *Survey* com acadêmicos da disciplina de Engenharia de Software no curso de Ciência da Computação e com especialistas na área de Engenharia de Software. Na seção 4 são apresentados os resultados obtidos na realização do Experimento Controlado com um grupo composto de acadêmicos e profissionais da área. Finalmente, na seção 5 são apresentadas as considerações finais do trabalho.

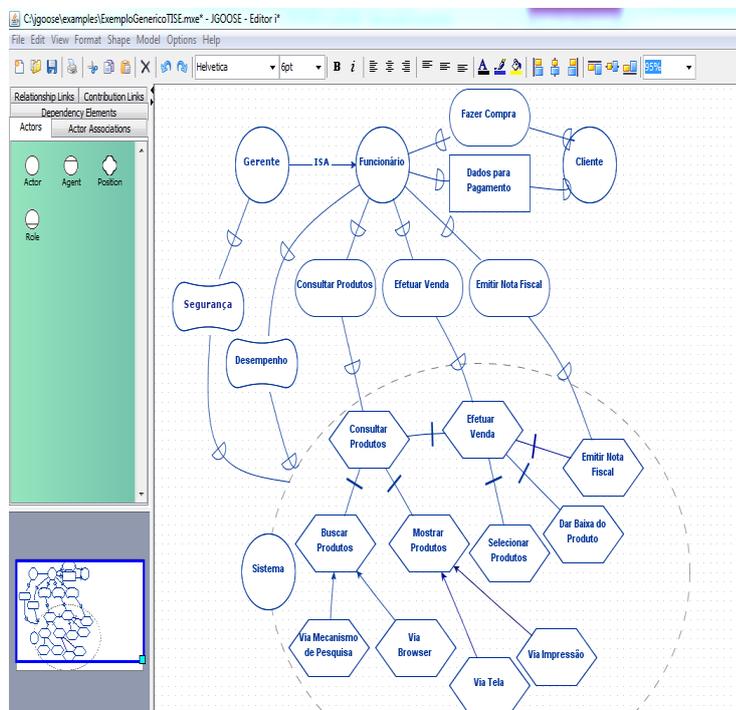
MATERIAIS E MÉTODOS

A ferramenta JGOOSE é um software capaz de auxiliar o mapeamento de modelos organizacionais i* para modelos funcionais na forma de Casos de Uso. Os modelos organizacionais são construídos via framework i* e geram os casos UML

a partir das diretrizes propostas em (Santander & Castro, 2002), sendo que casos de uso são representados na forma textual usando o *template* proposto em (Cockburn, 2001) e diagramas de Casos de uso conforme definido pela UML (Booch et al., 2005). Assim, com bases nos atores e nas suas dependências e razões estratégicas é possível gerar os casos de uso para um sistema computacional inserido nesse ambiente organizacional. A última versão do JGOOSE (Les, 2016) contém dois editores gráficos: o E4J i* (Editor for JGOOSE i*) (Merlim, 2015) e E4J Use Cases (Editor for JGOOSE Use Cases) (Peliser, 2014), sendo que os dois editores possuem a capacidade de gerar modelos organizacionais em i* e Casos de Uso, respectivamente.

O E4J i* permite a manipulação e criação de modelos i* denominados de Modelo de Dependências Estratégicas (SD) e Modelo de Razões Estratégicas (SR) convertendo estes modelos em uma estrutura do JGOOSE e da linguagem iStarML (Cares et al., 2008). Depois de construído, um modelo i* pode ser salvo em formato mxe. (nativo do editor), em formato de imagem ou exportado para a linguagem iStarML. A figura 1 apresenta a tela principal deste editor.

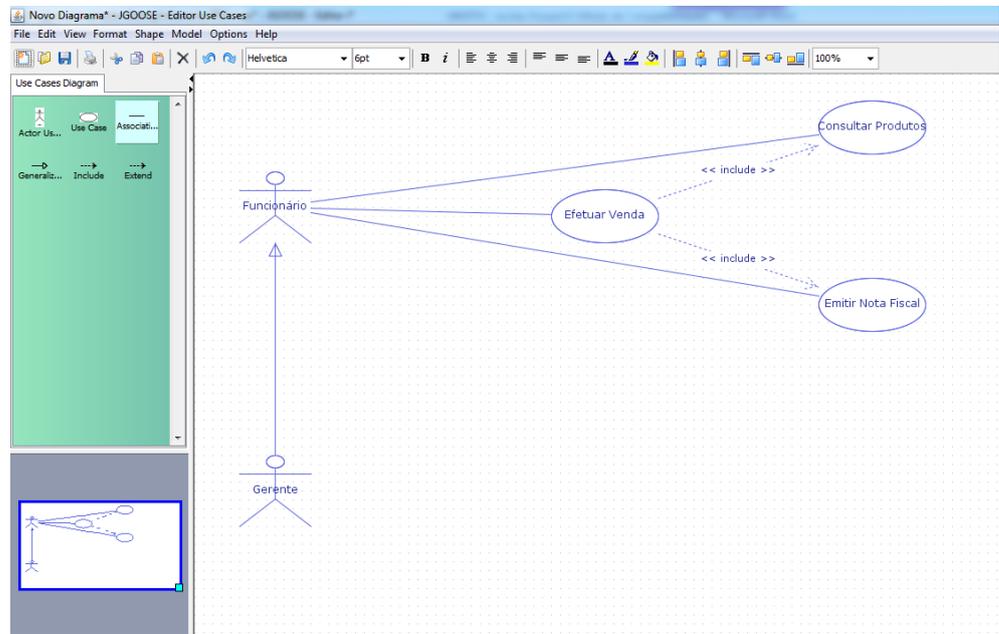
Figura 1 - Tela do editor gráfico E4J i* com um modelo exemplo



Já o E4J Use Cases permite editar diagramas de casos de uso auxiliando na criação diagramática de casos de uso e alteração de diagramas já gerados. Os casos de uso mapeados pelo JGOOSE servem como sustentação estrutural para os

diagramas gerados pelo E4J Use Cases. Também é possível salvar os diagramas gerados em diferentes formatos de imagem além do formato base .mxe, passível de edição posterior. A figura 2 apresenta a tela principal do editor E4J Use Cases.

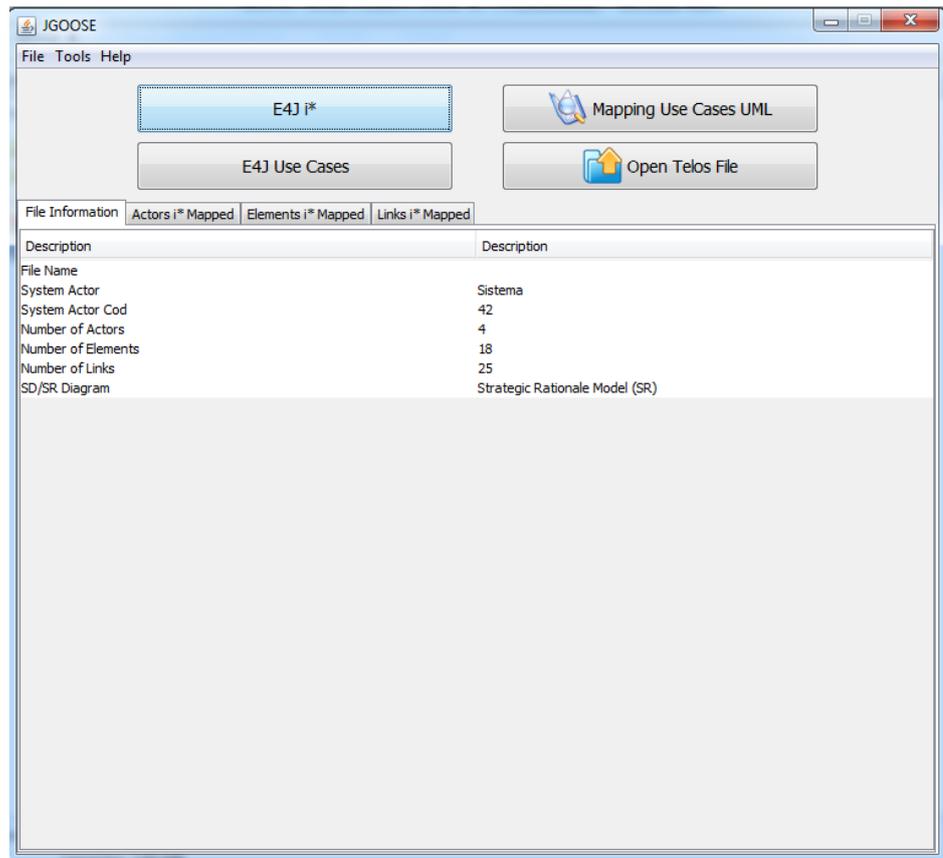
Figura 2 – Tela do Editor gráfico E4J Use Cases com um modelo exemplo



Desta forma, o ambiente JGOOSE que integra dois editores já mencionados, caracteriza-se por ser *standalone* sem precisar de outras ferramentas para derivar casos de uso a partir de modelos i*. A figura 3 apresenta a tela principal da ferramenta JGOOSE.

Por outro lado, a avaliação da JGOOSE passa pela utilização de princípios da Engenharia de Software Experimental. Esta área de estudo faz parte da Engenharia de Software. Seu objetivo é evoluir o conhecimento em Engenharia de Software a partir da aplicação de abordagem científica (experimentação) na construção de novos métodos e técnicas para apoio ao desenvolvimento de software (Travassos, 2002). O tipo de experimentação mais adequada depende dos resultados finais esperados, do objeto de estudo, ou mesmo das propriedades do processo de software utilizado durante a experimentação.

Figura 3 - Tela principal da ferramenta JGOOSE



Por a experimentação ser uma nova forma de tratamento na Engenharia de Software, o número de classificações é bem amplo. A literatura traz três estratégias principais para a experimentação, pois a classificação dos experimentos está sempre ligada ao princípio de estratégias empíricas, que podem ser elas:

Survey: Investigação executada em retrospectiva, que é conduzido quando algumas técnicas ou ferramentas já tenham sido utilizadas. Seus objetivos são de determinar peculiaridades ou características (descritivo), explicar o porquê dos desenvolvedores escolherem determinada técnica (explanatório) ou estudo precedente para uma investigação mais profunda (exploratório). Para a coleta de informações quantitativas e qualitativas é necessário o uso de questionários. O *Survey* é habilitado a levantar um grande número de variáveis para serem avaliadas, porém é necessário a redução destas variáveis para facilitar o processo de coleta de dados e classificação. O *Survey* não oferece nenhum meio possível de manipulação de variáveis nem controle sobre a execução ou medição.

Estudo de caso: Utilizado no monitoramento de atividades, projetos e atribuições, que se destinam à observação de uma qualidade específica,

estabelecendo relacionamento entre qualidades distintas. Mesmo com o número de controle sendo baixo, o estudo de caso difere do *Survey*, pois possui controle sobre a medição de variáveis. Por exemplo, a aplicação de pequenos números de casos em controle básico sobre as variáveis com resultados qualitativos. Seu maior problema é a possibilidade de fatores de confusão, sendo assim é difícil diferenciar o efeito proveniente de um fator do efeito proveniente de outro fator.

Experimento: frequentemente realizado em laboratório, proporcionando maior nível de controle. Tem objetivo principal de manipular uma ou mais variáveis mantendo outras fixas, avaliando o efeito resultante. Estes experimentos podem ser realizados in-vitro sob condições laboratoriais, ou in-vivo sob condições usuais. Os experimentos são adequados para a confirmação de teorias, conhecimento convencional, observar os relacionamentos, analisar as previsões dos modelos ou legitimar medidas. É importante salientar que os experimentos são geralmente relacionados à dois tipos de variáveis: dependentes e independentes. As variáveis independentes referem-se à entrada do processo de experimentação. Estas também se chamam "fatores" e apresentam a causa que afeta o resultado do processo de experimentação. O próprio valor de um fator se chama "tratamento". As variáveis dependentes referem-se à saída do processo de experimentação. Essas variáveis apresentam o efeito que é causado pelos fatores do experimento. O próprio valor de uma variável dependente se chama "resultado"(Travassos, 2002).

Tabela 1 – Fatores do Survey, Estudo de Caso e Experimento

Fator	Survey	Estudo de caso	Experimento
Controle de execução	Nenhum	Nenhum	Tem
Controle da medição	Nenhum	Tem	Tem
Controle da investigação	Baixo	Médio	Alto
Facilidade de repetição	Alta	Baixa	Alta
Custo	Baixo	Médio	Alto

Fonte: Tabela de nivelamento do Survey, Estudo de Caso e Experimento (Travassos, 2002).

Na tabela 1, apresenta-se um resumo das três metodologias estudadas. Pode-se observar as três metodologias em comparação com o controle de execução, medição, investigação, facilidade de repetição e o custo. Assim, é possível verificar que o Experimento é a metodologia mais precisa em resultados finais devido ao

controle de execução e coleta de dados. O controle total sobre o processo e as variáveis, e também a possibilidade de repetição é a maior vantagem do experimento. Por outro lado, o *Survey* possui a vantagem da facilidade de repetição podendo atingir níveis diferenciados de questionários em pouco tempo, gerando uma grande quantidade de dados as serem analisados. Finalmente o estudo de caso é o mais adequado diante de análises quantitativas resultando em dados de maior precisão, devido ao controle de medição. Contudo, devido à indisponibilidade de empresas para realizar um estudo de caso, esta metodologia não foi aplicada para avaliar a JGOOSE.

Desta forma, com base nos estudos mais aprofundados realizados sobre a engenharia de software experimental e mais especificamente sobre as metodologias propostas, detectou-se que é necessária uma avaliação da ferramenta JGOOSE no âmbito acadêmico e industrial. O intuito da avaliação é ter a disponibilidade argumentativa no futuro tanto na parte qualitativa como na parte quantitativa avaliada. Assim, optou-se por realizar avaliações em modo *Survey*, em nível acadêmico, devido à praticidade de repetição, facilidade de acesso a esse tipo de público e possibilidade de obter uma grande quantidade de resultados, o que permite uma análise mais abrangente. Assim, foi realizado um *Survey* com acadêmicos de uma disciplina de Engenharia de Software e outro, com pesquisadores especialistas da área. Este último foi realizado para obter informações mais precisas, críticas e baseadas em um alto nível de conhecimento desses pesquisadores no assunto. Os resultados destes estudos são apresentados na próxima seção.

Por outro lado, também detectou-se a necessidade de realizar um estudo avaliativo experimental no âmbito acadêmico, com o controle de algumas variáveis. As variáveis independentes (entradas para o experimento) são os elementos que podem ser utilizados para construir os modelos SD e SR via framework i^* . As variáveis dependentes são elementos que podem ser utilizados para criar os casos de uso. Estes elementos são dependentes pois são gerados a partir de um processo de mapeamento que utiliza os elementos dos modelos SD e SR como base. Ambos tipos de variáveis foram considerados com a intenção de obter informações mais detalhadas sobre a eficácia da JGOOSE bem como apontar possíveis erros no uso da ferramenta. O benefício dessa metodologia em nível acadêmico é o controle de variáveis independentes e dependentes, controle de

investigação, medição e execução. Destaca-se também pela facilidade de repetição. Maiores detalhes sobre este experimento são apresentados na seção 4.

RESULTADOS OBTIDOS COM A UTILIZAÇÃO DA METODOLOGIA SURVEY

Observando o processo de planejamento, execução e avaliação no uso da metodologia *Survey* proposto em (Travassos et al., 2002), foram realizadas duas avaliações da ferramenta JGOOSE. A primeira, envolveu estudantes da disciplina de PES I (Processo de Engenharia de Software I) do curso de Ciência da Computação, a qual é integralizada em 102h/a e possuía 25 alunos. As equipes que participaram do estudo consistiam em no máximo 3 integrantes totalizando 9 equipes. A segunda avaliação foi feita com especialistas na área da Engenharia de Software, sendo convidados 7 pesquisadores, porém obtendo a participação de apenas 3. Pode-se atribuir a baixa participação devido a falta de disponibilidade de alguns pesquisadores e/ou outras razões desconhecidas. Em ambas avaliações nenhuma variável foi controlada a fim de medir o desempenho da ferramenta JGOOSE, sobre as diferentes perspectivas (alunos/especialistas). Na avaliação com os estudantes, cada grupo possuía seus próprios modelos organizacionais i* revisados e utilizaram a ferramenta JGOOSE para gerar os casos de uso UML. Já na avaliação com os especialistas, estes receberam instruções escritas de como utilizar a ferramenta JGOOSE, não sofreram influência e as dúvidas em relação a funcionalidade da ferramenta foram sanadas via email, pessoalmente quando possível e/ou via skype.

Para cada avaliação foi elaborado um questionário, sendo que para os especialistas foram criadas 10 questões disponibilizadas via link exclusivo do aplicativo *SurveyPlanet* (Survey Planet, 2016) e para os estudantes foram geradas 7 questões disponibilizadas via moodle da disciplina. Assim, a coleta de dados é classificada como autoaplicável via internet, propiciando baixo custo, resultados rápidos e segurança. Outra vantagem dessa auto-aplicação é o fato de evitar confusão de fatores, pois a organização favorece dados mais precisos e contundentes.

Os *Surveys* realizados com os alunos e com os especialistas são classificados como explanatórios, sendo interseccionais pois a coleta de dados ocorreu em intervalo único de tempo, sendo a dos estudantes em 04 horas e dos especialistas

em 20 dias (os pesquisadores puderam responder o questionário sobre o JGOOSE do período de 09/07/2016 a 31/07/2016). Apesar dos questionários não terem controle de variáveis definidas, os mesmos foram direcionados a investigar a eficiência do JGOOSE (incluindo os editores gráficos E4J Use Cases e E4J i*) na obtenção de casos de uso UML a partir de modelos i*.

Os questionários foram desenvolvidos com questões atitudinais coletando assim informações sobre as atitudes e opiniões dos entrevistados e as perguntas foram do tipo *Open-ended*. Este tipo de pergunta permite maior liberdade ao entrevistado e ao mesmo tempo procura evitar respostas do tipo “sim” e “não”. Esta estratégia permitiu obter uma quantidade maior de informações com as questões elaboradas. No entanto, estas questões também defrontam os aplicadores do experimento (autores) com a difícil tarefa de realizar a análise qualitativa das respostas, o que envolve interpretação e análise para conclusões generalizáveis. Os resultados de ambos os *Surveys*, apresentando em termos percentuais o grau de satisfação dos entrevistados no uso da ferramenta, são descritos nas figuras 4 e 5. O gráfico da figura 4 exhibe as 7 perguntas feitas com os alunos e o gráfico da figura 5, as 10 perguntas feitas com especialistas da Engenharia de Software. Cabe ressaltar que as sete primeiras perguntas descritas a seguir, são as mesmas para ambos grupos alvo. Segue a síntese de cada pergunta para os dois grupos.

- Pergunta 1: “Você considera que os casos de uso gerados realmente são os principais para o sistema em questão?”

Síntese das respostas: sete grupos da disciplina em engenharia de software responderam que sim e duas equipes relataram que a ferramenta gerou incorretamente. Já no *Survey* com especialistas, todos responderam que os casos de uso gerados faziam sentido.

- Segunda pergunta: “Os casos de uso gerados são consistentes, fazem sentido?”

Síntese das respostas: as nove equipes relataram que eram consistentes e faziam sentido. A mesma resposta foi obtida de todos os especialistas.

- Terceira pergunta: “Como você procederia à busca de outros casos de uso importantes para o sistema? Por que eles não foram gerados pela ferramenta?”

Síntese das respostas: foi possível notar que segundo a maioria dos acadêmicos entrevistados, a ferramenta possui uma deficiência em gerar outros

casos de uso importantes para o sistema. Dois grupos argumentaram que outros casos de uso poderiam ser gerados se os modelos i* fossem melhorados e outra equipe apontou que a análise de documentos alternativos tais como estudo de viabilidade e documentos da empresa alvo poderia ajudar nesse processo. Outras três equipes relataram que a decomposição mais detalhada de cada dependência e ator no modelo SR, poderia levar a descoberta de novos casos de uso. Outras três equipes responderam que não encontraram outros casos de uso importantes. Uma equipe também acrescentou que o projeto não estava claro para a equipe e os modelos i* refletiam essa imaturidade, o que justificava que a ferramenta não estava equivocada em não gerar importantes casos de uso. Outra equipe também agregou que segundo sua visão, os outros casos de uso importantes estavam implicitamente incluídos em casos de uso já gerados pela ferramenta.

Por outro lado, um especialista afirmou que para buscar outros casos de uso seria necessário conhecer melhor o domínio, outro relata que seria necessário especificar um modelo até um limite aceitável de complexidade e o último afirmou que após o mapeamento, complementaria os casos de uso obtidos. Em relação a segunda parte da pergunta, um especialista apontou que a ferramenta falha em encontrar novos casos de uso particularmente em situações específicas e os outros afirmaram que a modelagem i* foca nas dependências gerais e, portanto, algumas dependências importantes podem não ter sido incluídas.

- Quarta pergunta: “A descrição textual e em particular os cenários principal e secundário dos casos de uso gerados automaticamente pela ferramenta são úteis, fazem sentido? Eles precisam ser modificados complementados?”

Síntese das respostas: a ferramenta mostrou uma pequena falha desagradando 34% dos grupos entrevistados e fazendo sentido para 66% dos grupos entrevistados. Para todos especialistas sim fazem sentido e são úteis. Apenas um especialista relata que o analista tem que complementar os detalhes.

- Quinta pergunta: “O fato de poder modificar e salvar as descrições textuais dos casos de uso facilita o teu trabalho?”

Síntese das respostas: novamente a ferramenta satisfaz a maioria dos grupos de acadêmicos da disciplina, sendo que 89% responderam que a ferramenta JGOOSE facilita o trabalho de um engenheiro de requisitos. Para todos os especialistas as descrições textuais dos casos de uso fazem sentido e facilitam o trabalho.

- Sexta pergunta: “O fato de poder editar o diagrama de casos de uso no editor E4J Use Cases facilita o teu trabalho de analista?”

Síntese das respostas: outra vez a ferramenta apresentou eficácia com seu editor integrado E4J Use Cases pois 89% dos grupos de acadêmicos respondeu que facilitou o trabalho como analista. Para todos os especialistas, o editor E4J Use Cases facilita o trabalho de analista.

- Sétima pergunta “O fato de poder editar os modelos organizacionais i* e gerar uma nova versão do Diagrama de Casos de Uso facilita o teu trabalho de analista?”

Síntese das respostas: oito equipes (89%) responderam que sim e uma equipe (11%) argumentou que não, demonstrando mais uma vez o bom desempenho da ferramenta JGOOSE. Para todos os especialistas o fato de poder editar os modelos organizacionais i* facilita o trabalho analista.

As perguntas que seguem, foram destinadas apenas aos especialistas da área da engenharia de Software.

- Oitava pergunta “Existem vantagens/desvantagens em derivar casos de uso usando esta abordagem?”.

Síntese das respostas: Para todos os especialistas existem vantagens e desvantagens, porém nas respostas obtidas, há mais vantagens do que desvantagens. Dentre as vantagens obtidas cabe destacar: a possibilidade de aproveitar a modelagem organizacional gerada, o contexto organizacional é considerado útil para gerar requisitos funcionais e não-funcionais de sistemas computacionais, a abordagem permite ampliar a compreensão do sistema como um todo bem como a possibilidade de complementar ou identificar falhas dos casos de uso e a automação provida pela ferramenta JGOOSE. Entre as desvantagens apontadas cabe salientar: dedicação de tempo à modelagem organizacional para que os modelos sejam consistentes o que exige um bom nível de conhecimento da técnica i*, a perda de requisitos não-funcionais no mapeamento, o fato de poder complementar os casos de uso na parte diagramática e descritiva pode ocasionar uma inconsistência com os modelos SD/SR e a não integração com outros artefatos do processo.

- Nona pergunta: “Existem problemas em relação à ferramenta JGOOSE e editores E4J i* e E4J Use Cases integrados à mesma e quais seriam as sugestões de melhoria?”:

Síntese das respostas: Para os especialistas entrevistados há sim pequenos problemas em relação aos editores tais como a falta de *tooltips* associados aos elementos da ferramenta, atualização do editor i* para incorporar a versão da linguagem i* 2.0, manter a consistência dos modelos, evolução da ferramenta com a integração de novos artefatos, entre outros.

- Décima pergunta “Qual sua opinião geral sobre a ferramenta JGOOSE?”

Síntese das respostas: Todos os especialistas ficaram satisfeitos em usufruir da ferramenta elaborada em âmbito acadêmico.

Figura 4 – Gráfico das Respostas Obtidas com os Estudantes em Porcentagem de Eficiência

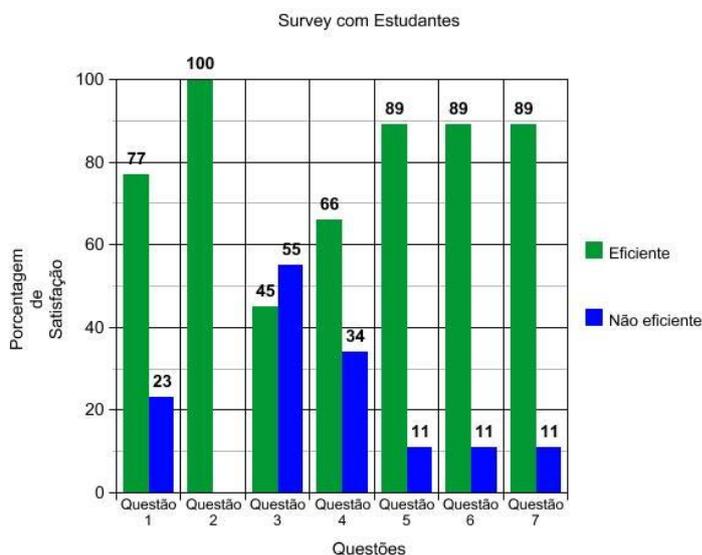
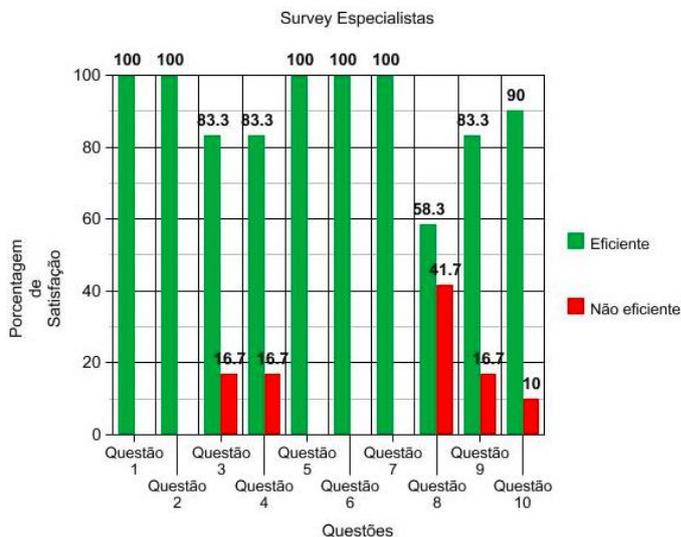


Figura 5 – Gráfico das Respostas Obtidas com os Especialistas em Porcentagem de Eficiência



RESULTADOS OBTIDOS COM A REALIZAÇÃO DO EXPERIMENTO CONTROLADO

Em termos práticos, o objetivo de um experimento na engenharia de software pode incluir mostrar que um particular método ou ferramenta é superior, de alguma maneira, a outro. Pode-se desejar mostrar que, para um particular elemento sob investigação, diferentes condições ambientais ou de qualidade dos recursos, podem afetar o uso ou os resultados de uma investigação (Freire, 2015). Neste contexto, cabe ressaltar que o objetivo principal do experimento foi realizar uma análise de algumas variáveis independentes e outras dependentes associadas ao JGOOSE e editores E4J i* e E4J Use Cases. Mais especificamente, no editor E4J i*, foram consideradas como variáveis independentes os elementos dos modelos SD e SR utilizados pelos participantes para construir os modelos i* que seriam utilizados para gerar os casos de uso. Estes elementos são: *Actors (Actor, Role, Agent e Position)*, *Actors Associations (ISA, Is-part-of, Plays, Covers, Occupies e Instance of)*, *Dependency Elements (Goal, Task, SoftGoal e Resource)*, *Contributions links (Make, Some+, Help, Break, Some-, Hurt, Unknown, And e Or)* e *Relationship Links (Dependency, Means-end e task-decomposition)*.

Por outro lado, os elementos que compõe os casos de uso UML foram considerados como variáveis dependentes já que foram gerados a partir dos modelos SD e SR em i* e editáveis no E4J Use Cases. As variáveis dependentes consideradas foram: *Actor Uses Cases, Use Case, Association, Generalization, Include e Extend*. Assim, ambos tipos de variáveis foram analisadas e verificadas pelos participantes e pesquisador controlando o experimento.

O experimento foi realizado em laboratório junto a uma equipe de 5 integrantes. Dois desses integrantes possuem curso superior em Ciência da Computação e atuam em empresas de desenvolvimento de software, dois outros são acadêmicos do último do curso de Ciência da Computação e um integrante, apesar de ser acadêmico, já atua em âmbito empresarial. Todos os integrantes tiveram 8 horas durante uma semana de aprendizado da ferramenta e após este período, estipulou-se um mês de experimento, sendo dividido em 10 horas para a avaliação experimental do editor gráfico E4J i* pois este contém uma maior quantidade de elementos a serem avaliados e 8 horas para a avaliação experimental do editor gráfico E4J Use Cases. Para a coleta de dados foram alocadas 8 horas. Assim, o JGOOSE foi avaliado em todas as suas funcionalidades.

Cada participante montou seus próprios modelos i^* e utilizou a ferramenta JGOOSE para derivar os casos de uso. Assim, foi atribuída a cada participante do experimento, a tarefa de verificar os dois editores gráficos da ferramenta e, por fim, a ferramenta JGOOSE como um todo. Ao final da avaliação cada pesquisador atribuiu uma nota geral sobre a ferramenta e uma nota para cada editor gráfico integrado à ferramenta (E4J use cases e E4J i^*). Além das notas atribuídas representadas na tabela 2, cada participante justificou descritivamente cada nota. Assim, o experimento possibilitou obter duas categorias de avaliações: uma quantitativa e outra qualitativa advinda das respostas de cada participante.

Tabela 02 que representa notas avaliativas de 0 a 10 de cada participante

	Âmbito	E4J Use	E4 i^*	JGOOSE
Participante 1	Acadêmico	8	8	9
Participante 2	Acadêmico	9	8	8
Participante 3	Profissional	7	7	7
Participante 4	Profissional	7	7	7
Participante 5	Acadêmico/Profiss	8	7	8

As respostas dos participantes 1 e 2 são semelhantes, sendo que ambos relataram que consideraram a ferramenta excelente no que tange a ambos editores pois os mesmos possibilitavam modificar os diagramas i^* e Casos de Uso, visando acrescentar outros elementos importantes que não haviam sido obtidos no primeiro mapeamento realizado pela ferramenta. Na prática, segundo esses participantes, isto possibilita melhorar ambos os modelos gerados visando torná-los mais completos e úteis para o engenheiro de requisitos. O participante 2 relatou que há uma certa dificuldade em dominar a técnica/modelagem i^* e, assim, é necessário um tempo maior para realizar um bom diagrama na ferramenta. Além desta argumentação, o participante ressaltou que em algumas vezes, as descrições textuais dos cenários principal e secundário são confusas devido à falta de detalhes. O participante 1 também defende o fato de que estando em um âmbito acadêmico, a ferramenta auxilia muito bem quem pretende fazer um diagrama e encontrar os casos de uso de um sistema computacional pretendido. Os dois participantes descreveram que após alterar o

diagrama de casos de uso gerado, o modelo SD/SR utilizado não era alterado para refletir as mudanças, ocasionando uma inconsistência entre ambos diagramas.

Os participantes 3 e 4 fazem parte do ambiente empresarial e avaliaram a ferramenta em nível de desempenho profissional em um possível uso nas empresas nas quais trabalham. Para o pesquisador 3, em geral a ferramenta é boa, porém a técnica i* é pouco conhecida e não está em prioridade no ambiente comercial, acarretando que o seu uso exige um tempo para entender e dominar a técnica. Porém, considera que a ferramenta ajuda no processo de entendimento do ambiente organizacional e derivação de funcionalidades dos sistema pretendido, principalmente porque os editores apresentam boa usabilidade, o que melhora o desempenho nas tarefas realizadas. Os principais erros detectados no estudo realizado por esse participante foram atribuídos a falhas no uso da técnica i* ocasionados pela falta de conhecimento, complexidade da própria técnica e construção de modelos muito simples com poucos detalhes. O participante 4 também apontou as mesmas vantagens sobre a ferramenta descritas pelo pesquisador 3, porém ressaltou que a ferramenta ainda necessita de melhorias pois em algumas situações, é necessário poder rever o sistema de forma mais prática após a geração dos casos de uso, o que facilitaria encontrar possíveis erros. Também foi destacado pelo participante a perda de requisitos não-funcionais no mapeamento realizado pela JGOOSE.

O participante 5 realizou o experimento controlado sob ambas perspectivas acadêmica e profissional, ressaltando que a ferramenta sim possui um alto nível para a área acadêmica, porém para a área comercial, na qual são desenvolvidos sistemas de maior complexidade e tamanho, o editor E4J i* possui algumas deficiências que precisam ser corrigidas tais como: ao utilizar o zoom sobre uma dependência de um elemento que estiver curvado, ao retornar do zoom, esta dependência modifica-se de tamanho tornando sua visualização prejudicada, o que desagrada visualmente o usuário. Além disso, ao fechar a janela de aplicação do editor SD/SR, o programa se encerra. Segundo esse participante, nesse caso, a aplicação deveria voltar à janela principal da JGOOSE.

Cabe ressaltar que para todos os cinco participantes a ferramenta apresenta um bom desempenho por ser *Standalone* e utilizar a técnica i*.

CONSIDERAÇÕES FINAIS

Neste artigo são apresentados os resultados de avaliações realizadas da ferramenta JGOOSE, a qual permite derivar casos de uso UML a partir de modelos organizacionais i*. Para avaliar essa ferramenta, foram utilizadas duas metodologias de experimentação propostas na Engenharia de Software Experimental. A metodologia *Survey* foi aplicada a dois grupos diferentes, um de estudantes e outro de especialistas na área de Engenharia de Software. Após a análise das respostas dos questionários, foi possível notar que a ferramenta foi considerada satisfatória pela maioria de ambos grupos. Cabe ressaltar que o questionário aplicado foi fator decisivo durante a avaliação pois permitiu que os entrevistados exibissem uma opinião crítica e construtiva dentro do contexto abordado, o que gerou em uma grande quantidade de dados a serem analisados. Entre as principais limitações da pesquisa está a baixa participação de entrevistados, particularmente de especialistas da área bem como o direcionamento das respostas ocasionado pelas questões pré-elaboradas.

Em um segundo momento, foi utilizada a metodologia Experimento Controlado. Este experimento permitiu coletar as diferentes visões de cada categoria de participante (acadêmicos e profissionais) em um ambiente controlado. O fato de cada participante verificar as mesmas variáveis resultou na obtenção de diferentes tipos de opiniões em determinados contextos e situações diferenciadas. Assim, cada participante exibiu uma análise quantitativa e outra qualitativa da ferramenta JGOOSE e dos editores E4J Use Case e E4J i*. Cabe ressaltar que em sua maioria, os participantes relataram o alto nível eficiência da ferramenta em âmbito acadêmico bem como algumas melhorias necessárias para utilizar a ferramenta em âmbito empresarial.

Como trabalhos futuros pretende-se utilizar a outra metodologia denominada de estudo de caso para avaliar a ferramenta JGOOSE bem como realizar as melhorias nessa ferramenta sugeridas pelos participantes do *Survey* e do experimento controlado.

Evaluation of the jgoose tool using the experimental software engineering

ABSTRACT

Evaluate computational tools as well as new techniques or processes require special care in the planning, execution and analysis on the performed experiment. In this context, the experimental software engineering presents some methodologies that can be useful in conducting experiments. This paper describes the results obtained from the Survey and Controlled Experiment methodologies used to evaluate the JGOOSE tool, which was developed by our research group and can automatically derive UML use cases from i* organizational models. The Survey was applied to students of the software engineering course as well as some to experts on the software engineering. The Controlled Experiment was applied to a group of five participants involving students and professionals of the Computer Science.

KEYWORDS: computational tool; i* (istar); survey; controlled experiment.

REFERÊNCIAS

- BOOCH, G. RUMBAUGH, J. JACOBSON, I. UML: Guia do Usuário, 2ª Edição, Editora Campus, 2005.
- COCKBURN, A. Writing effective use cases. [S.l.]: Addison-Wesley Reading. 2001.
- CARES, C., Franch, X., Perini, A., Susi, A. istarmI: An xml-based model interchange format for i*. In: Proc. 3rd Int. i* Workshop, Recife, Brazil. [S.l.: s.n.], v. 322, p.1316. 2008.
- LINAKEK, J.; SULAMAN, M. S; HOST, M.; MELLO, M. R; **Guidelines for Conducting Surveys in Software Engineering v 1.1**. Department of Computer Science Lund University, Sweden, Experimental Software Engineering Group Systems Engineering Computer Science Department COPPE/ UFRJ, Rio de Janeiro – RJ: June, 2015.
- LES. Laboratório de Engenharia de Software. Disponível em <<http://www.inf.unioeste.br/~les/>>, acesso 15/08/2016.
- FREIRE, M. A; **Formalização de Experimentos Controlados Em Engenharia de Software**. Tese de Doutorado. Universidade Federal Do Rio Grande Do Norte, Centro de Ciências Exatas e Da Terra Programa de Pós-Graduação Em Sistemas e Computação. Natal – RN: Março, 2015.
- MERLIM, L. P. ; SILVA, A. L. B. ; SANTANDER, V. F. A ; SILVA, I. F. ; CASTRO, J. F. B. . **Integrating the E4J editor to the JGOOSE tool**. In: XVIII Workshop de Engenharia de Requisitos (WER), Universidad Ricardo Palma, Lima, 2015.
- PELISER, D. **E4J Use Cases: um editor de diagrama de casos de uso integrado à ferramenta JGOOSE**. Cascavel – PR: [s.n.], 2014.
- SANTANDER, V. F. A., CASTRO, J. F. B. Deriving Use Cases from Organizational Modeling. In: IEEE Joint International Requirements Engineering Conference - RE, 2002.
- Survey Planet. Disponível em <<https://surveyplanet.com/>>, acesso em 17/08/2016.
- TRAVASSOS, G. H; GUROV, D; AMARAL, E. A. G. **Introdução à Engenharia de Software Experimental**. Rio de Janeiro – RJ, 2002.
- VICENTE, A. A.; SANTANDER, V. F. A; CASTRO, J. B; FREITAS DA SILVA, I; REYS M; FRANCISCO G; **JGOOSE: A Requirements Engineering Tool to Integrate i***

Organizational Modeling with Use Cases in UML. Ingeniare. Revistachilena de ingeniería, 2009. 6–20 p.

WOHLIN, C; HÖST, M; REGNELL, B; RUNESSON, P; OHLSSON, M. C; WESSLÉN, A;
Experimentation in Software Engineering. Springer Heidelberg New York Dordrecht London, 2000. 12–18 p.

YU, E. S. K; **Modelling Strategic Relationships For Process Reengineering.** Tese (Doutorado) – University of Toronto, 1995.

Recebido: 26 ago. 2016.

Aprovado: 23 nov. 2017.

DOI:

Como citar: Avaliando a ferramenta jgoose utilizando princípios da engenharia de software experimental. R. Eletr. Cient. Inov. Tecnol, Medianeira, v. 8, n. 15, 2017. E – 4562.
Disponível em: <<https://periodicos.utfpr.edu.br/recit>>. Acesso em: XXX.

Correspondência:

Direito autoral: Este artigo está licenciado sob os termos da Licença Creative Commons-Atribuição 4.0 Internacional.

